

Research Article

An Image Enhancement Method Using the Quantum-Behaved Particle Swarm Optimization with an Adaptive Strategy

Xiaoping Su,¹ Wei Fang,² Qing Shen,³ and Xiulan Hao³

¹ School of Computer & Software Engineering, Nanjing Institute of Industry Technology, Nanjing 210046, China

² School of IOT Engineering, Jiangnan University, Wuxi, Jiangsu 214122, China

³ School of Information Engineering, Huzhou Teachers College, Huzhou, Zhejiang 313000, China

Correspondence should be addressed to Xiaoping Su; xiaoping.su@yahoo.cn

Received 25 January 2013; Accepted 22 April 2013

Academic Editor: Jui-Sheng Lin

Copyright © 2013 Xiaoping Su et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Image enhancement techniques are very important to image processing, which are used to improve image quality or extract the fine details in degraded images. In this paper, two novel objective functions based on the normalized incomplete Beta transform function are proposed to evaluate the effectiveness of grayscale image enhancement and color image enhancement, respectively. Using these objective functions, the parameters of transform functions are estimated by the quantum-behaved particle swarm optimization (QPSO). We also propose an improved QPSO with an adaptive parameter control strategy. The QPSO and the AQPSO algorithms, along with genetic algorithm (GA) and particle swarm optimization (PSO), are tested on several benchmark grayscale and color images. The results show that the QPSO and AQPSO perform better than GA and PSO for the enhancement of these images, and the AQPSO has some advantages over QPSO due to its adaptive parameter control strategy.

1. Introduction

Image enhancement is one of the important image techniques in low-level image processing, the purpose of which is to improve the quality of an image for machine analysis or visual perception of human being [1]. For grayscale image enhancement, the most widely used method is histogram equalization which is based on the assumption that a uniformly distributed grayscale histogram has the best visual contrast [1, 2]. The degraded images with poor contrast can be enhanced by stretching the dynamic range of the grayscale histogram, namely, by grayscale rescaling [3]. In this method, a linear or nonlinear transform between the degraded image and enhanced image is needed. Generally speaking, various types of nonlinear transforms are needed to enhance various types of degradations. An acceptable approach is to use a set of generalized transforms, which are generally combinations of the basic types of nonlinear transforms. Tubbs used a regularized incomplete Beta function as the generalized transform described by a set of parameters, and consequently the image enhancement problem is reduced to a global optimization problem [4].

Grayscale image enhancement, however, cannot be generalized to color image enhancement directly. Several factors, including selection of the color space, characteristics of the human visual system, and color contrast sensitivity, are not taken into consideration in grayscale image enhancement but have to be all treated in color image enhancement [5–7]. The most popular hardware-oriented model for color images is the RGB (red, green, blue) model that has been widely used in color monitors, color video cameras, and computer multimedia applications. If grayscale image enhancement is applied directly to the three components (R, G, B) of a degraded color image, they may be prone to produce color artifacts which look very strange for human beings [1, 5–8]. Another color space is HIS which stands for three main attributes: hue, intensity, and saturation, generally used to distinguish one color from another [6, 7]. Using the HIS color space in color image processing system can decouple the achromatic and chromatic information. Moreover, this color space is close to the color perceiving properties of the human visual system. Therefore, according to the employed color space, the existing color image enhancement techniques can be classified into three categories: the techniques using

the RGB color space [5, 7], the techniques using the LHS color space [2, 9, 10], and other methods [11, 12]. In this work, the color image enhancement approach is based on HIS color space.

The goal of this paper is to propose two novel objective functions for the estimation of the parameters of the regularized incomplete Beta functions, which in turn are used for grayscale and color image enhancement, respectively. The objective functions are optimized by quantum-behaved particle swarm optimization (QPSO).

The QPSO algorithm, as a variant of PSO, was inspired by quantum mechanics and trajectory analysis of the individual particle's behavior in PSO [13–19]. The particle in QPSO needs no velocity vector but sampling its position directly through a double exponential distribution. Except for the population size, the only algorithmic parameter is the contraction-expansion (CE) coefficient which should be adjusted to balance the global search and the local search of the algorithm. Thus, an efficient control strategy for the CE coefficient can enable the algorithm to perform generally well when QPSO is applied to real-world problems. In this paper, we propose an adaptive control method for the CE coefficient and used the improved QPSO for the enhancement of grayscale and color images, using two proposed new objective functions for estimating the parameters of the Beta functions.

The rest of the paper is organized as follows. Sections 2 and 3 address the concepts and the proposed methods of grayscale and color image enhancement, respectively. Section 4 provides a brief introduction of the QPSO algorithm, followed by the proposal of the improved QPSO with the adaptive control strategy. The experimental results for grayscale and color image enhancement are given in Section 5. The paper is concluded in Section 6.

2. Grayscale Image Enhancement

2.1. Grayscale Rescaling. Generally speaking, the purpose of grayscale image enhancement is to improve the low-level image quality, that is, to highlight some specified features or recover the significant details in degraded images. One of the common characteristics of degraded images is poor contrast. The degraded images with poor contrast can be enhanced by stretching the dynamic range of the grayscale histogram, that is, by grayscale rescaling. The simplest rescaling transform is given by

$$I_{xy}^* = f(i_{xy}), \quad (1)$$

where i_{xy} and I_{xy}^* are the grayscale values of the pixel at (x, y) within the input image and the output enhanced image, respectively, and f is a linear or nonlinear transform. Four typical types of widely used nonlinear transforms for grayscale image enhancement are shown in Figure 1, where vertical axes and abscissa stand for the gray value after and before enhancement processing, respectively. Generally, the input range of a pixel should be first normalized into the range $[0, 1]$, and thus (1) is rewritten as

$$i_{x,y} = i_{\min} + (i_{\max} - i_{\min}) \times T(n(i_{x,y})), \quad (2)$$

where i_{\min} and i_{\max} are the minimum and maximum values of $i_{x,y}$, respectively, $n(i_{x,y})$ normalizes $i_{x,y}$ into the range $[0, 1]$, and $T(\cdot)$ ($0 \leq T(\cdot) \leq 1$) is one kind of nonlinear transforms.

In general, since the types of degradations for an input image are usually unknown and a particular type of nonlinear transforms can only enhance a particular type of degradation, we need various types of nonlinear transforms, with each described by a set of parameters, to enhance various types of degradation. An acceptable approach is to use a set of generalized transforms which combine the four types of degradation transforms [20]. In this study, we adopt the method proposed by Tubbs, who used the regularized incomplete Beta functions to simulate the four types of transforms as shown in Figures 1(a)–1(d) [4]. The standard Beta density function and the regularized incomplete Beta function are, respectively, given by

$$f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}, \quad (3)$$

$$F(x, \alpha, \beta) = B^{-1}(\alpha, \beta) \times \int_0^x t^{\alpha-1}(1-t)^{\beta-1} dt,$$

where $0 \leq x \leq 1$, $\alpha, \beta > 0$, and $B(\alpha, \beta)$ is the Beta function defined by

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1} dt. \quad (4)$$

Therefore, the generalized transform can be given by

$$T(i_{xy}) = f(i_{xy}, \alpha, \beta) = \int_0^{i_{xy}} \frac{t^{\alpha-1}(1-t)^{\beta-1}}{B(\alpha, \beta)} dt, \quad (5)$$

where $i_{x,y}$ denotes the gray value in (x, y) , $0 \leq i_{x,y} \leq 1$, and $0 \leq T(i_{x,y}) \leq 1$.

2.2. Generalized Transformation of Gray Level. Consequently, if the gray level of pixel (x, y) is $f(x, y)$, which turned into $f'(x, y)$ after enhancement, the process of generalized transformation consists of the following steps.

Step 1. Normalize the gray value of each pixel by

$$u(x, y) = \frac{[f(x, y) - i_{\min}]}{(i_{\max} - i_{\min})}, \quad (6)$$

where i_{\min} and i_{\max} are the minimum and maximum values, respectively, of the input image and $0 \leq u(x, y) \leq 1$.

Step 2. Transform the input image by the regularized incomplete Beta function $T(i_{xy})$ in (5):

$$u'(x, y) = T(u(x, y)), \quad (7)$$

where $u'(x, y) \in [0, 1]$ is the output image.

Step 3. Calculate the gray value of the output image:

$$f'(x, y) = (i_{\max} - i_{\min}) \times u'(x, y) + i_{\min}. \quad (8)$$

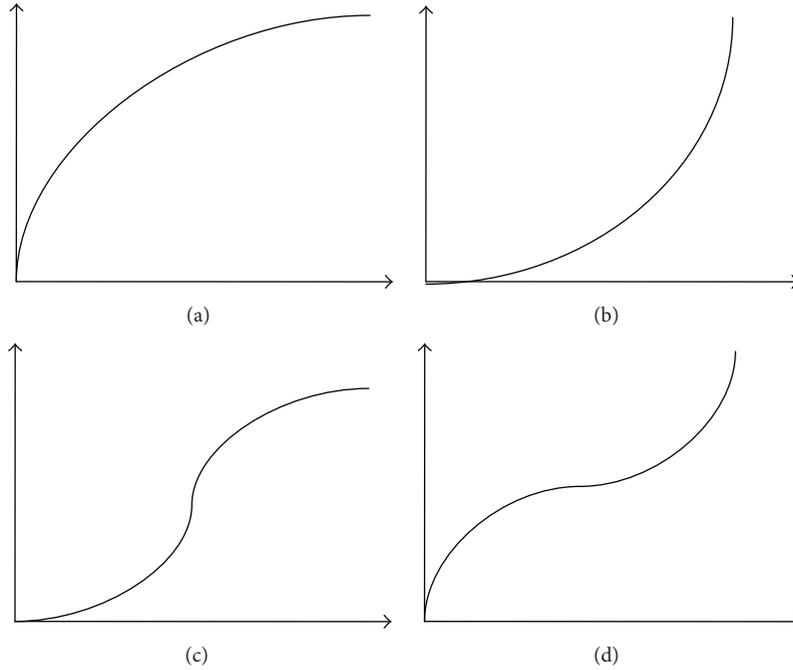


FIGURE 1: Four types of nonlinear transforms for grayscale image enhancement: (a) a transform stretching dark regions; (b) a transform stretching lighter regions; (c) a transform stretching middle and compressing two ends; (d) a transform compressing middle and stretching two ends.

2.3. Objective Function. In this paper, the image enhancement problem is converted into an optimization so that it is important to determine the objective (fitness) function which is used to evaluate the goodness of solution to the problem. According to the characteristics in grayscale image enhancement, we proposed a fitness function which is formed by combining the AC power measure (F_{ac}) and the Brenner's measure (F_{br}). Both F_{ac} and F_{br} are used to evaluate the focus accuracy of a grayscale image [21].

For a given grayscale image $M \times N$ in size, the AC power measure is defined by

$$F_{ac} = \frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N i_{xy}^2 - \left(\frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N i_{xy} \right)^2, \quad (9)$$

where i_{xy} denotes the gray value in (x, y) and $n (= M \times N)$ is the total number of pixels within the image. It is obvious from the above equation that if the F_{ac} value is increased, the average pixel variance of the image is increased and the overall contrast of the image will be improved accordingly.

The Brenner's measure F_{br} is given by

$$F_{br} = \sum_{x=1}^{M-2} \sum_{y=1}^N (i_{x,y} - i_{x+2,y})^2. \quad (10)$$

F_{br} is the total square differences between each pixel and the corresponding pixel located two pixels away. We can found from (10) that an image with good contrast will have a large F_{br} value.

With the definitions of F_{ac} and F_{br} , the fitness function for grayscale image enhancement is given by

$$\text{Fitness} = \frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N f'^2(x, y) - \left[\frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N f'(x, y) \right]^2 + \frac{1}{n} \sum_{x=1}^{M-2} \sum_{y=1}^N [f'(x, y) - f'(x+2, y)]^2, \quad (11)$$

where $f'(x, y)$ is the pixel value of (x, y) after generalized transformation. Therefore, the goal of grayscale image enhancement is to find a proper parameter set (α, β) such that $f'(x, y)$ obtained after generalized transformation minimizes the above fitness function.

3. Color Image Enhancement

As mentioned in Section 2, for grayscale image enhancement, the general method is histogram equalization or its variations. However, grayscale image enhancement cannot be generalized to color image enhancement directly. Several factors, such as selection of the color space, characteristics of the human visual system, and color contrast sensitivity, need not to be taken into consideration in grayscale image enhancement, but should all be treated in color image enhancement [5–7].

The most popular hardware-oriented model is the RGB (red, green, blue) model used in color monitors, color video cameras, and computer multimedia applications. Histogram equalization and its variants are quite useful in emphasizing

TABLE 1: The estimated parameters corresponding to the best fitness values obtained by GA, PSO, QPSO, and AQPSO.

Algorithms	α	β	The best fitness value
GA	9.1202346	9.8338221	6.4565e + 007
PSO	9.8950179	9.8950179	6.6369e + 007
QPSO	9.9999367	9.9999387	6.6685e + 007
AQPSO	9.9999832	9.9999958	6.6714e + 007

TABLE 2: Mean best fitness and standard deviation over 30 trial runs of GA, PSO, QPSO, and AQPSO.

Algorithms	Mean fitness value	Standard deviation
GA	6.4294374e + 007	3.8293278e + 005
PSO	6.4809664e + 007	7.7541706e + 005
QPSO	6.5876416e + 007	1.6503350e + 006
AQPSO	6.7193757e + 007	1.5930591e + 006

important details in grayscale images; nevertheless, if applied individually to the three components (R, G, B) of a degraded color image, they would destroy the natural color balance in the scene. Image enhancement approaches applied directly to the three components (R, G, B) of a degraded color image may easily produce color artifacts which may make the enhanced image look very strange for human beings frequently. Thus, it is inappropriate for the human visual system to apply color image enhancement directly in the RGB color space [1, 5–8].

The characteristics (three main attributes) generally used to distinguish one color from another are hue, intensity, and saturation (HIS). Using the HIS color space in color image processing system can decouple the achromatic and chromatic information. Besides, this color space is close to the color perceiving properties of the human visual system. Therefore, the proposed color image enhancement approach in this paper is applied in the HIS color space.

Here it is assumed that each input degraded color image is originally represented in the RGB color space and is converted into the HIS color space for enhancement. When an input degraded color image in the HIS color space is obtained, it is necessary to normalize the input range of a pixel into $[0, 1]$ and then determine a set of generalized transforms for the degraded color image.

The regularized incomplete Beta function in (5) $T(i_{xy})$ is used as the generalized transform to transform the intensity of the pixel (x, y) , that is,

$$i'(x, y) = T(i(x, y)), \quad (12)$$

where $i'(x, y) \in [0, 1]$. Then the intensity $f'(x, y)$ of the output image can be given by

$$f'(x, y) = (i_{\max} - i_{\min}) \times i'(x, y) + i_{\min}. \quad (13)$$

Because the information in the hue component H has no clear application to image enhancement [11] based on the experiments obtained in this study, we do not attempt any form of hue processing. That is, in the proposed approach the hue component H in the HIS color space is left unchanged.

The last characteristic, saturation component S should be homogenized first in this approach.

After the pretreatment mentioned above, we can come into the main processing of color image enhancement. In the proposed enhancement for color image, the fitness function is formed by three performance parameters, namely, the AC power measure (F_{ac}), the Brenner's measure (F_{br}), and the information-noise change measure (IN_c).

Similar to (9) for a grayscale image, the AC power of a given color image $M \times N$ in size is defined by

$$F_{ac} = \frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N i_{xy}^2 - \left(\frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N i_{xy} \right)^2, \quad (14)$$

where i_{xy} is the pixel value at (x, y) and $n (= M \times N)$ is the total number of pixels within the image. The Brenner's measure F_{br} is given by

$$F_{br} = \sum_{x=1}^{M-2} \sum_{y=1}^N (i_{x,y} - i_{x+2,y})^2, \quad (15)$$

which shares the same meaning as that in (10). IN_c is given by

$$IN_c = \sum_{n(h)>t} 1, \quad (16)$$

where $n(h)$ is the number of pixels in the input histogram with level h and t is a threshold. Here IN_c is an auxiliary measure which can prevent the effect of overthresholding (resulting in some important details to be dropped), or overenhancing of noise.

In this work, the fitness function $f(\cdot)$ for color image enhancement is given by

$$\text{Fitness} = IN_c \times F_{ac} + F_{br}$$

$$= \left(\sum_{n(h)>t} 1 \right) \times \left(\frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N f'^2(x, y) - \left[\frac{1}{n} \sum_{x=1}^M \sum_{y=1}^N f'(x, y) \right]^2 \right)$$

$$+ \sum_{x=1}^{M-2} \sum_{y=1}^N [f'(x, y) - f'(x+2, y)]^2, \quad (17)$$

from which we can see that the larger the fitness function value is, the better the enhanced color image will be.

4. QPSO and QPSO with Adaptive Strategy

4.1. The QPSO Algorithm. The particle swarm optimization (PSO) method is a member of a wider class of Swarm Intelligence methods used for solving GO problems, which

was originally proposed by Kennedy and Eberhart as a simulation of social behavior of bird flock and was initially introduced in [22, 23]. Since the origin of PSO, there has been a lot of work done in the improvement and applications of the algorithm [24–33]. In PSO, the particles represent the candidate solutions to the problem, fly through a multidimensional search space, and evaluate their position to a goal at every iteration as well as sharing memories of “best” position to find out the optima or suboptima finally. Since its origin, the PSO algorithm has already been used in a large variety of application areas due to its easy implementation and low requirement in memory and CPU speed. However, as proved by Van Den Bergh [34], PSO is not a global convergence-guaranteed algorithm because the particles are restricted to a finite search space at each iteration, which weakens the global search ability of the algorithm and makes the PSO algorithm easily trap into the local optima. In order to overcome this shortage, Sun et al. proposed a novel variant of PSO, named quantum-behaved particle swarm optimization (QPSO), in which each individual particle has quantum behavior [13, 14]. As has already been proved in theory and several applications, QPSO has a better performance compared with PSO both in precision and efficiency since the sampling space of each particle at each iteration can cover the whole feasible space of the problem at hand [35–44]. In this paper, we introduce an adaptive mechanism into the QPSO and therefore propose an adaptive QPSO (AQPSO).

In the original PSO with M individuals, each individual is treated as a volumeless particle in the N -dimensional space, with the position vector and velocity vector of particle i at n th iteration represented as $X_{i,n} = (X_{i,n}^1, X_{i,n}^2, \dots, X_{i,n}^N)$ and $V_{i,n} = (V_{i,n}^1, V_{i,n}^2, \dots, V_{i,n}^N)$. Vector $P_{i,n} = (P_{i,n}^1, P_{i,n}^2, \dots, P_{i,n}^N)$ is the best previous position (the position giving the best objective function value since initialization of the population) of particle i and is called personal best (*pbest*) position, and vector $G_n = (G_n^1, G_n^2, \dots, G_n^N)$ is the position of the best particle among all the particles in the population found so far and is known as global best (*gbest*) position. Without loss of generality, we consider the following minimization problems:

$$\begin{aligned} & \text{Minimize } f(X), \\ & \text{s.t. } X \in S \subseteq R^n, \end{aligned} \quad (18)$$

where $f(X)$ is an objective function and S is the feasible space. Accordingly, $P_{i,n}$ can be updated by

$$P_{i,n} = \begin{cases} X_{i,n} & \text{if } f(X_{i,n}) < f(P_{i,n-1}) \\ P_{i,n-1} & \text{if } f(X_{i,n}) \geq f(P_{i,n-1}). \end{cases} \quad (19)$$

Since G_n is the personal best position giving the best fitness value among all particles, it can be found by

$$G_n = P_{g,n}, \quad (20)$$

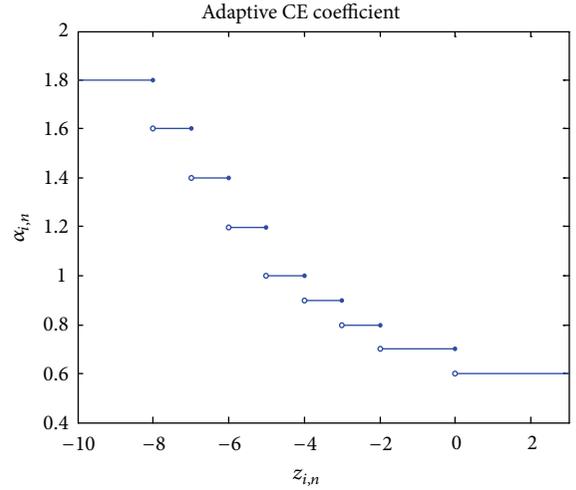


FIGURE 2: The adaptive CE coefficient for each particle.

where $g = \arg \min_{1 \leq i \leq M} [f(P_{i,n})]$. With the above definitions, each particle in the original PSO updates its velocity and position by the following equations:

$$V_{i,n+1}^j = V_{i,n}^j + c_1 r_{i,n}^j (X_{i,n}^j - P_{i,n}^j) + c_2 R_{i,n}^j (X_{i,n}^j - G_n^j), \quad (21)$$

$$X_{i,n+1}^j = X_{i,n}^j + V_{i,n+1}^j, \quad (22)$$

for $i = 1, 2, \dots, M; j = 1, 2, \dots, N$, where c_1 and c_2 are called acceleration coefficients. The parameters $r_{i,n}^j$ and $R_{i,n}^j$ are two different random numbers distributed uniformly on $(0, 1)$, that is, $r_{i,n}^j, R_{i,n}^j \sim U(0, 1)$. Generally, the value of $V_{i,n}^j$ is restricted in the interval $[-V_{\max}, V_{\max}]$.

In [45], Clerc and Kennedy undertook the first formal analysis of the particle's trajectory and the stability properties of the PSO algorithm. They simplified the particle swarm to a second-order linear dynamical system whose stability depends on the system poles or the eigenvalues of the state matrix. Their analyses essentially revealed the fact that the particle swarm may converge if each particle converges toward its local attractor, $p_{i,n} = (p_{i,n}^1, p_{i,n}^2, \dots, p_{i,n}^N)$ defined at the coordinates

$$p_{i,n}^j = \frac{c_1 r_{i,n}^j P_{i,n}^j + c_2 R_{i,n}^j G_n^j}{c_1 r_{i,n}^j + c_2 R_{i,n}^j}, \quad 1 \leq j \leq N, \quad (23)$$

or

$$p_{i,n}^j = \phi_{i,n}^j P_{i,n}^j + (1 - \phi_{i,n}^j) G_n^j, \quad (24)$$

where $\phi_{i,n}^j = c_1 r_{i,n}^j / (c_1 r_{i,n}^j + c_2 R_{i,n}^j)$ with regard to the random numbers $r_{i,n}^j$ and $R_{i,n}^j$ in (21) and (23). The acceleration coefficients c_1 and c_2 in the original PSO are generally set to be equal, that is, $c_1 = c_2$ so that $\phi_{i,n}^j$ is a sequence of uniformly distributed random number over $(0, 1)$. As a result, (24) can be restated as

$$p_{i,n}^j = \phi_{i,n}^j P_{i,n}^j + (1 - \phi_{i,n}^j) G_n^j, \quad \phi_{i,n}^j \sim U(0, 1). \quad (25)$$

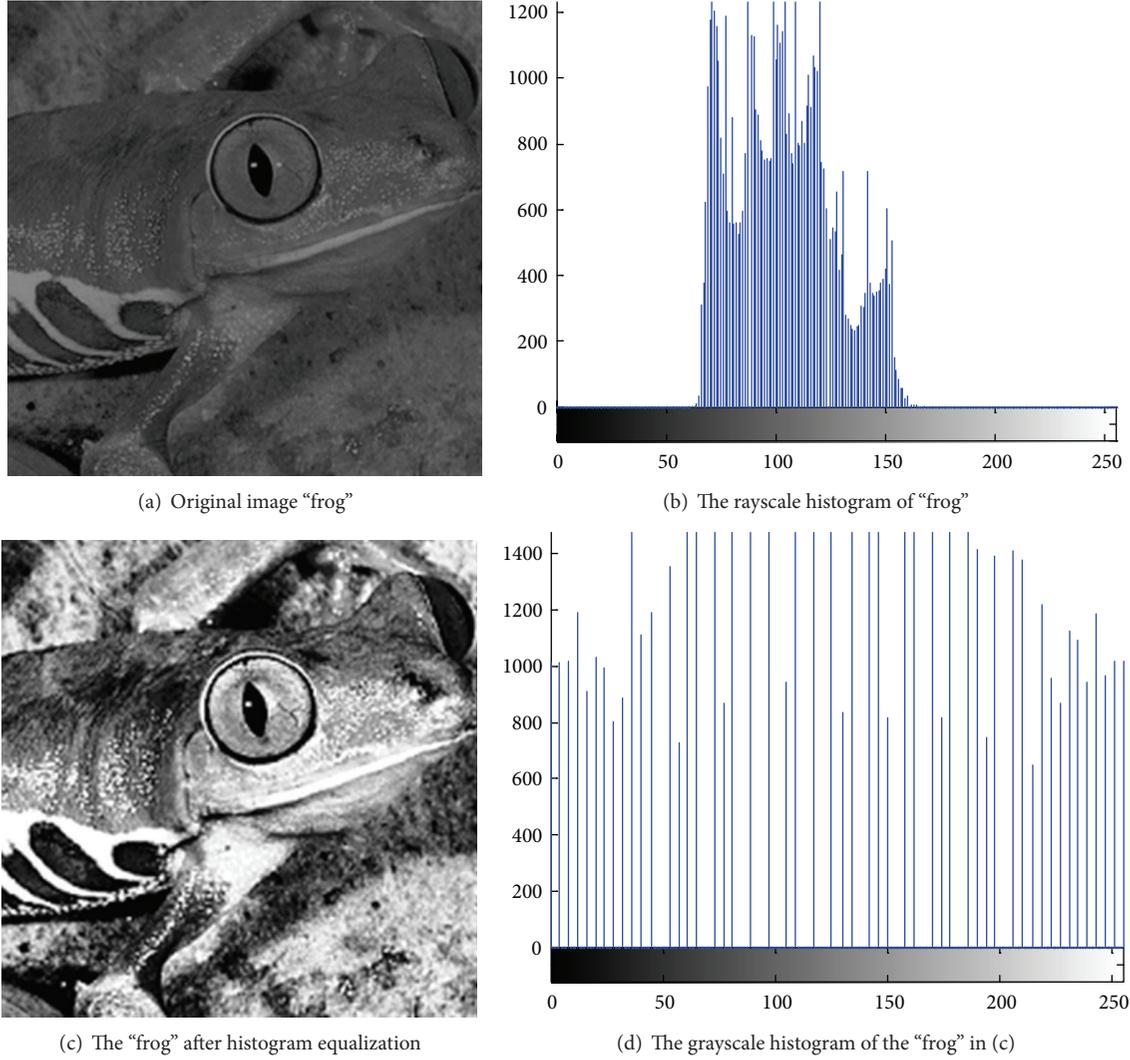


FIGURE 3: The image "frog" and its histogram equalization image, as well as their grayscale histograms.

The above equation indicates that $p_{i,n}$ is a stochastic point that lies in a hyperrectangle, with $P_{i,n}$ and G_n being two ends of its diagonal, and that moves following $P_{i,n}$ and G_n . In the process of convergence, the particle moves around and careens toward point p_i with its kinetic energy (velocity) declining to zero, like a returning satellite orbiting the earth. As such, the particle in PSO can be considered as the one flying in an attraction potential field centered at point $p_{i,n}$ in Newtonian space. It has to be in bound state for the sake of avoiding explosion and guaranteeing convergence. If these conditions are generalized to the case that the particle in PSO moves in quantum space, it is also indispensable that the particle moves in a quantum potential field to ensure the bound state. From the perspective of quantum mechanics, the bound state in quantum space, however, is entirely different from that in Newtonian space, which leads to the proposal of the QPSO algorithm.

In QPSO, each single particle is assumed to be a spinless one with quantum behavior. Thus state of the particle is characterized by wavefunction ψ , where $|\psi|^2$ is the probability

density function of its position. At the n th iteration, particle i moves in N -dimensional space with a δ potential well centered at $p_{i,n}^j$ on j th dimension for $1 \leq j \leq N$. Let $Y_{i,n+1}^j = |X_{i,n+1}^j - p_{i,n}^j|$; we can get the following normalized wavefunction at iteration $n+1$ satisfying the bound condition that $\psi(Y_{i,n+1}^j) \rightarrow 0$ as $Y_{i,n+1}^j \rightarrow \infty$:

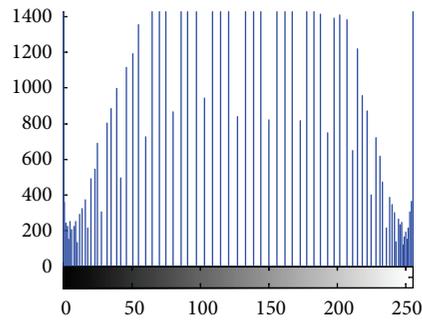
$$\psi(Y_{i,n+1}^j) = \frac{1}{\sqrt{L_{i,n}^j}} \exp\left(-\frac{Y_{i,n+1}^j}{L_{i,n}^j}\right), \quad (26)$$

where $L_{i,n}^j$ is the characteristic length of the wavefunction. By the definition of wavefunction, the probability density function is given by

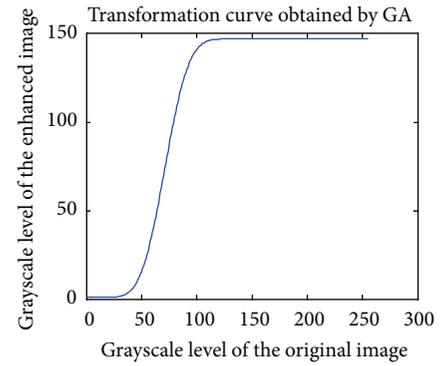
$$Q(Y_{i,n+1}^j) = |\psi(Y_{i,n+1}^j)|^2 = \frac{1}{L_{i,n}^j} \exp\left(\frac{-2Y_{i,n+1}^j}{L_{i,n}^j}\right), \quad (27)$$



(a1) The "frog" enhanced by GA



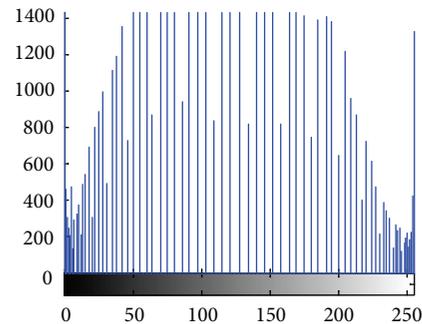
(b1) The grayscale histogram of the "frog" in (a1)



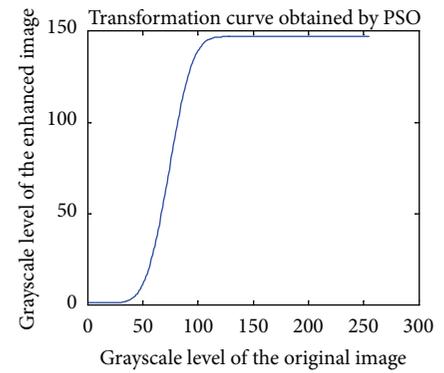
(c1) Transformation curve of GA



(a2) The "frog" enhanced by PSO



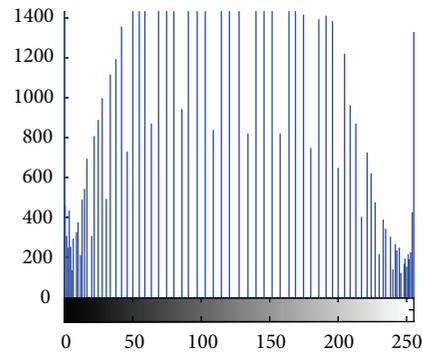
(b2) The grayscale histogram of the "frog" in (a2)



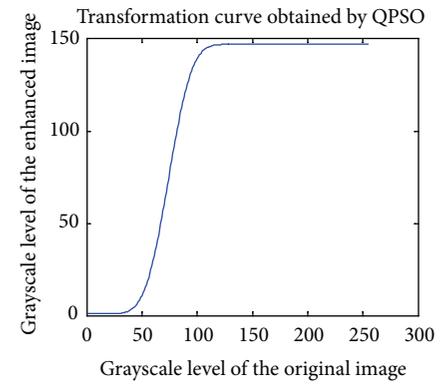
(c2) Transformation curve of PSO



(a3) The "frog" enhanced by QPSO



(b3) The grayscale histogram of the "frog" in (a3)

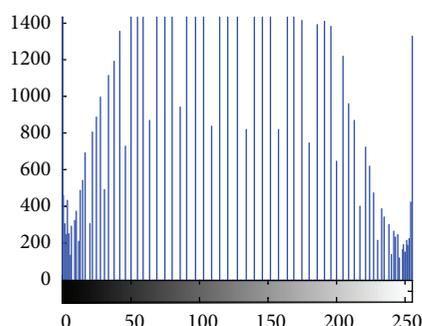


(c3) Transformation curve of QPSO

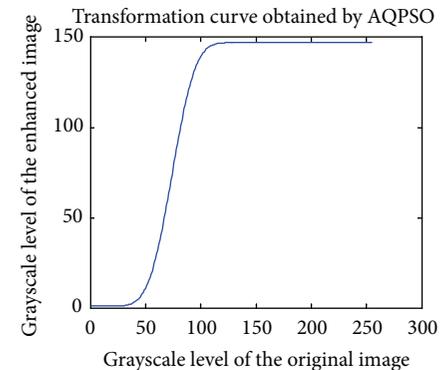


(a4) The "frog" enhanced by AQPSO

(a)



(b)



(c)

FIGURE 4: The comparison of enhancement using GA, PSO, QPSO, and AQPSO for "frog."



FIGURE 5: Comparison of enhancements using different approaches for "Lenna".

and thus the probability distribution function is

$$F(Y_{i,n+1}^j) = 1 - \exp\left(\frac{-2Y_{i,n+1}^j}{L_{i,n}^j}\right). \quad (28)$$

Using Monte Carlo method, we can obtain the j th component of position particle i at the $(n + 1)$ th iteration by

$$X_{i,n+1}^j = p_{i,n}^j \pm \frac{L_{i,n}^j}{2} \ln\left(\frac{1}{u_{i,n+1}^j}\right), \quad u_{i,n+1}^j \sim U(0, 1), \quad (29)$$

where $u_{i,n}^j$ is a sequence of random numbers uniformly distributed in $(0, 1)$. As proposed in [14], the value of $L_{i,n}^j$ is given by

$$L_{i,n}^j = 2\alpha |X_{i,n}^j - C_n^j|, \quad (30)$$

where $C_n = \bar{P}_n = (C_n^1, C_n^2, \dots, C_n^N)$ is called mean best ($mbest$) position defined by the mean of the $pbest$ positions of all particles and thus $C_n^j = \bar{P}_n^j = (1/M) \sum_{i=1}^M P_{i,n}^j$ ($1 \leq j \leq N$). Therefore the position of the particle updates as follows:

$$X_{i,n+1}^j = p_{i,n}^j \pm \alpha |X_{i,n}^j - C_n^j| \cdot \ln\left(\frac{1}{u_{i,n+1}^j}\right). \quad (31)$$

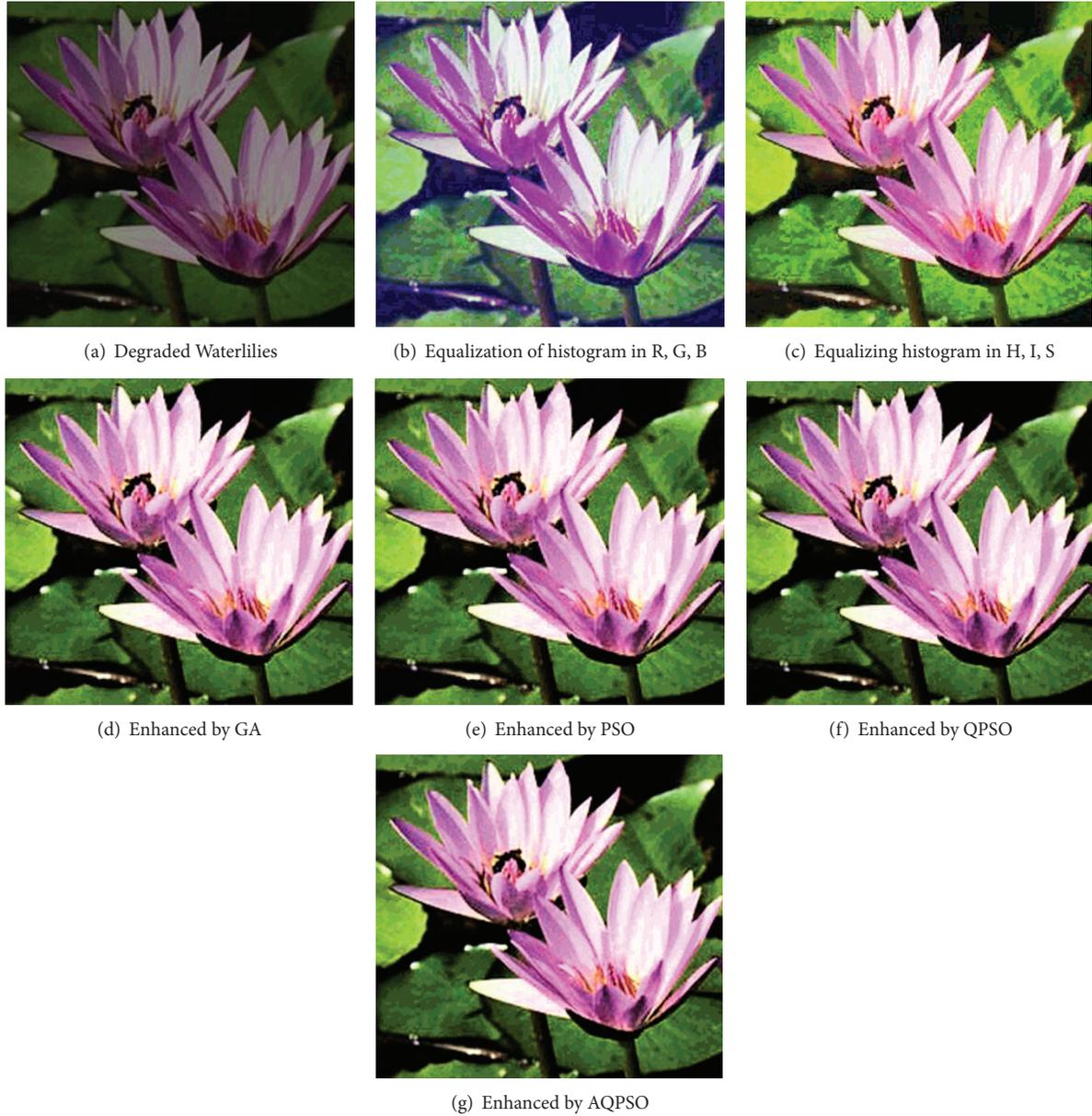


FIGURE 6: Comparison of enhancements using different approaches for “Waterlilies”.

The parameter α in (30) or (31) is known as the contraction-expansion (CE) coefficient, which can be tuned to control the convergence speed of the algorithms. The PSO with (31) is known as quantum-behaved particle swarm optimization (QPSO), the procedure of which is outlined in Algorithm 1, where $\text{rand } i(\cdot)$, $i = 1, 2, 3$, are separately generated random numbers uniformly distributed in $(0, 1)$.

4.2. QPSO with Adaptive Strategy. It has been proved that the CE coefficient α must be set as $\alpha < 1.781$ to guarantee the boundedness of the particles so that the whole particle swarm may converge [15]. Whenever α is set to be larger than $\alpha > 1.781$, the particles will explode. As a result, it is necessary to set α smaller than 1.781 for the purpose of convergence when

the algorithm is being applied to real-world problems. In [15], two methods of controlling α were analyzed comprehensively. When fixed-valued α is used, it is suggested that fixing the value of α as 0.75 during the search process can generate good algorithmic performance in general. On the other hand, it has been verified that decreasing α linearly from α_0 to α_1 ($\alpha_0 > \alpha_1$) can yield generally good algorithmic performance.

In this paper, we propose an adaptive control method for the CE coefficient by using the following error function:

$$\Delta F_{i,n} = \frac{f(P_{i,n}) - f(G_n)}{\min(|f(P_{i,n})|, |f(G_n)|)}, \quad (32)$$

where $f(P_{i,n})$ is the fitness values of the $pbest$ position of particle i and $f(G_n)$ is the fitness value of the $gbest$ position

```

Begin
Initialize the population;
While (termination condition = false)
Do
  Compute mean best position  $C$ ;
  Properly select the value of  $\alpha$ ;
  for ( $i = 1$  to  $M$ )
    Evaluate the objective function value  $f(X_i)$ ;
    Update  $P_i$  and  $G$ ;
    For  $j = 1$  to  $N$ 
       $\varphi_{i,n}^j = \text{rand1}(\cdot)$ ;
       $P_{i,n}^j = \varphi_{i,n}^j P_{i,n}^j + (1 - \varphi_{i,n}^j) G_n^j$ ;
       $u_{i,n}^j = \text{rand2}(\cdot)$ ;
      if ( $\text{rand3}(\cdot) < 0.5$ )
         $X_{i,n+1}^j = P_{i,n}^j + \alpha |X_{i,n}^j - C_n^j| \ln(1/u_{i,n}^j)$ ;
      else
         $X_{i,n+1}^j = P_{i,n}^j - \alpha |X_{i,n}^j - C_n^j| \ln(1/u_{i,n}^j)$ ;
      end if
    end for
  end for
end do
end

```

ALGORITHM 1: Procedure of the QPSO.

at the n th iteration, respectively. $\min(x_1, x_2)$ returns the minimum of x_1 and x_2 . Equation (32) measures the proximity of a particle to the $gbest$ position. It is obvious that the smaller the value of $\Delta F_{i,n}$, is the closer the particle's $pbest$ position to the $gbest$ position, and accordingly the narrower the search range of the particle, leading the particle to loss of global search ability. If all the particles' $pbest$ positions cluster to the $gbest$ position, that is, the average value of $\Delta F_{i,n}$ over all the particles is small, the whole particle swarm may encounter premature convergence, since the search scope of the particle is determined by the multiplication of the CE coefficient and the distance from its current position to the mean best position. If the particles cluster together around the $gbest$ position, a larger CE coefficient α can prevent them from further clustering. Therefore, we propose an adaptive tuning method for α to help enhance the global search ability of the QPSO algorithm.

In the proposed adaptive QPSO (AQPSO) algorithm, each particle has its own CE coefficient during each iteration, instead of sharing the CE coefficient with the other particles. Generally, the value of α for the particle which is far away from the $gbest$ position has a larger value, while that for the particle close to the $gbest$ position has a smaller one that can make the particle fly toward the $gbest$ position more rapidly. Here, the proximity of the particle to the $gbest$ position is defined by the logarithmic value of $\Delta F_{i,n}$, namely,

$$\begin{aligned}
 z_{i,n} &= \ln(\Delta F_{i,n}) \\
 &= \ln(f(P_{i,n}) - f(G_n)) \\
 &\quad - \ln(\min(|f(P_{i,n})|, |f(G_n)|)).
 \end{aligned} \tag{33}$$

The CE coefficient for each particle is set according to the value of $z_{i,n}$ using the following piecewise function:

$$\alpha_{i,n}(z_{i,n}) = \begin{cases} 0.6, & z_{i,n} > 0, \\ 0.7, & -2 < z_{i,n} \leq 0, \\ 0.6 + 0.1 \times k, & -k - 1 < z_{i,n} \leq -k, \quad (k = 2, 3, 4), \\ 1.0 + 0.2 \times (k - 4), & -k - 1 < z_{i,n} \leq k, \quad (k = 5, 6, 7), \\ 1.8, & z_{i,n} \leq -8. \end{cases} \tag{34}$$

The graph of the adaptive CE coefficient is shown in Figure 1. When $z_{i,n} \leq -8$, the value of $\alpha_{i,n}$ is larger than 1.781, which means that the particle is in explosion and flies away rapidly from the $gbest$ position. This piecewise function for the value of the CE coefficient means that if the distance between the particle and the $gbest$ position is large, the CE coefficient should be set small to make the particle converge fast; if the distance is small, the CE coefficient can be large so that the particle can converge slowly to perform global search. This adaptive control method for the CE coefficient helps to maintain the diversity of the swarm at a certain level and ensures the global search ability of the algorithm during the whole search process (see Figure 2).

5. Experimental Results

5.1. Results for Grayscale Image Enhancement. The experiments on enhancement of the degraded grayscale image "frog" were performed by using GA, PSO, QPSO, and AQPSO, respectively. The "frog" is 256×256 in size. Figures 3(a) and 3(b) show its image and grayscale histogram,

TABLE 3: Unpaired *t*-test between the performances of the algorithms.

	Std. Error	<i>T</i> value	<i>P</i> value	Significance
AQPSO versus QPSO	418785.661	3.1456	0.0026	Very Significant
AQPSO versus PSO	323476.373	7.3702	<0.0001	Extremely Significant
AQPSO versus GA	299136.246	9.6925	<0.0001	Extremely Significant
QPSO versus PSO	332910.260	3.2043	0.0022	Very Significant

TABLE 4: Experimental results of each algorithm on “Lenna” and “Waterlilies”.

	GA	PSO	QPSO	AQPSO
Best fitness value				
Lenna	1.2679e + 003	1.4694e + 003	1.4770e + 003	1.4801e + 003
Waterlilies	1.2006e + 003	1.2406e + 003	1.2513e + 003	1.2605e + 003
Mean fitness value				
Lenna	1.2594e + 003	1.4324e + 003	1.4527e + 003	1.4737e + 003
Waterlilies	1.1915e + 003	1.1936e + 003	1.2045e + 003	1.2152e + 003
Standard deviation				
Lenna	3.5129e + 001	4.0285e + 001	4.4002e + 001	3.3019e + 001
Waterlilies	1.0017e + 001	1.0046e + 001	1.1691e + 001	1.0470e + 001

respectively, from which we can see that the contrast is poor and the details in skin and background are not clear. The most popular method for grayscale image enhancement is known to be histogram equalization, by which the image “frog” enhanced is shown in Figure 3(c). The corresponding grayscale histogram in Figure 3(d) indicates that the frequency of each gray level is the same, the entropy of the image now is the biggest, and the information combined is the largest. Nevertheless, it was pointed in [16] that the same frequency for each gray level is not bounded to result in the best visual effects, since the robustness of a grayscale image is directly related to the distribution of its gray level.

In the experiments for GA, the population size was set to be 60, the crossover and mutation probabilities were 0.6 and 0.01, respectively, and 20 binary bits composed a chromosome. For PSO, the inertia weight decreased linearly from 0.9 to 0.4 during the search process, and the two acceleration coefficients were fixed at 2.0. For QPSO, the CE coefficient decreased linearly from 1.0 to 0.5 in the course of search. The population sizes for all PSO variants, including PSO, QPSO, and AQPSO, were set to be 30. The maximum number of iterations for a single run of each algorithm was 500. Every algorithm was performed for 30 trial runs, and mean best fitness and standard deviation were recorded for performance comparison.

The estimated parameters α and β corresponding to the best fitness value obtained over 30 trial runs are listed in Table 1, which shows that the best fitness values obtained by QPSO and AQPSO are better than those by PSO and GA. Between QPSO and AQPSO, the latter appears to estimate the parameters more accurately than the former. In order to evaluate the average performance of each competitor algorithms, we recorded in Table 2 the mean best fitness values and standard deviations over 30 trial runs of GA, PSO, QPSO, and AQPSO. It can be observed that AQPSO obtained the mean best fitness value over 30 runs and is the

best among all the tested algorithms and that the second best performing algorithms is QPSO. Table 3 lists the *t* test results between the AQPSO and the other algorithms, showing that the outperformance of the AQPSO over its competitors is significant. Figure 4 shows the enhanced images by different approaches based on the proposed objective function in (11) for grayscale image, as well as their grayscale histograms and transformation lines. The enhanced image by each algorithm was yielded through the transforms with the parameters α and β estimated by the algorithm. Obviously, the image texture that was enhanced by any of GA, PSO, QPSO, or AQPSO is nicer than that in Figure 3(c). However, the best fitness obtained by AQPSO is larger than that by any other competitors, implying that AQPSO had a better performance than the others.

5.2. Results for Color Image Enhancement. Two color images, “Lenna” and “Waterlilies,” are used as benchmarks to evaluate the performance of the QPSO and AQPSO algorithms in color image enhancement. The two images are originally represented in the RGB color space, 256 * 256 in size. Two simple approaches for color image enhancement that equalize the histograms in R, G, B and histograms in H, I, S, respectively, were also employed for performance comparison. In the experiments for GA, population size was set to be 60, crossover operator and mutation operator were 0.6 and 0.01, respectively, 20 binary bits composes a chromosome, and moreover, the maximum of iterations was 500. In the experiments for PSO, QPSO, and AQPSO, each algorithm used 30 particles and lasted for a maximum of iterations of 500. Each of the tested population-based optimization algorithms performed 30 trial runs on each color image, with the mean fitness values and the standard deviation being recorded in Table 3.

It is shown in Table 4 that the mean fitness values obtained by AQPSO are slightly larger than those by PSO

TABLE 5: Unpaired t test between the performances of the algorithms for “Lenna” and “Waterlilies”.

	Std. error	T value	P value	Significance
Lenna				
AQPSO versus QPSO	10.044	2.0908	0.0409	Significant
AQPSO versus PSO	9.510	4.3429	<0.0001	Extremely significant
AQPSO versus GA	8.802	24.3465	<0.0001	Extremely significant
Waterlilies				
AQPSO versus QPSO	2.865	3.7343	0.0004	Extremely significant
AQPSO versus PSO	2.649	8.1535	<0.0001	Extremely significant
AQPSO versus GA	2.646	8.9586	<0.0001	Extremely significant

TABLE 6: The estimated parameters for Lenna corresponding to the best fitness values obtained by GA, PSO, QPSO, and AQPSO.

Algorithms	α	β	The best fitness value
GA	4.9213	9.7468	$1.2679e + 003$
PSO	4.9584	9.8429	$1.4694e + 003$
QPSO	4.9601	9.9998	$1.4770e + 003$
AQPSO	4.9617	9.9999	$1.4801e + 003$

TABLE 7: The estimated parameters for Waterlilies corresponding to the best fitness values obtained by GA, PSO, QPSO, and AQPSO.

Algorithms	α	β	The best fitness value
GA	2.7725	9.7258	$1.2006e + 003$
PSO	2.8831	9.8259	$1.2406e + 003$
QPSO	2.9421	9.9999	$1.2513e + 003$
AQPSO	2.9458	9.9998	$1.2605e + 003$

and GA, even than that by QPSO, which means that the AQPSO-based color image enhancement approach had the best performance among all the competitors. The significance of the AQPSO's outperformance is verified by t test as shown in Table 5. Tables 6 and 7 list the estimated parameters α and β corresponding to the best fitness value obtained over 30 trial runs. It is shown that the best fitness values obtained by QPSO and AQPSO are better than those by PSO and GA, implying that QPSO and AQPSO were able to estimate the parameters more accurately, and that AQPSO had better performance than QPSO as can be observed. Figures 5 and 6 show the enhanced images for “Lenna” and “Waterlilies” using different approaches based on the proposed objective function in (17) for color image. The enhanced image by each algorithm was yielded through the transforms with the parameters α and β listed in Tables 5 and 6. As can be observed in Figures 5 and 6, the image texture that was enhanced by any of GA, PSO, QPSO, or AQPSO is nicer than those by equalization of histogram in R, G, B or in H, I, S.

6. Conclusion

This paper proposed a novel objective function to evaluate the effectiveness of grayscale image enhancement based on the normalized incomplete Beta function and modified this

objective for color image enhancement in HIS space. Furthermore, the AQPSO algorithm which employs an adaptive control method for the CE coefficient was proposed to optimize the objective function for obtaining a proper set of parameters of the Beta function and achieving enhancement of a grayscale or color image.

AQPSO, along with QPSO, PSO, and GA, was tested on some well-known benchmark images for performance comparison on image enhancement. To do so, two groups of experiments were performed, one for the enhancement of a grayscale image and the other for the enhancement of a color image. The experimental results showed that QPSO and AQPSO were able to estimate the parameters more accurately and thus had better performance for both grayscale and color image enhancement than their competitor. Moreover, AQPSO generated better enhanced images than the original QPSO, as can be observed from the results.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 61103051), by the National Natural Science Foundation of China (Grants no. 61170119 and 61105128), by Natural Science Foundation of Jiangsu Province, China (Grant no. BK2010143), by the Ministry of Education Research in the humanities and social sciences planning fund (Grant no. 12YJAZH120), and by the Zhejiang Provincial Natural Science Foundation of China (Grants no. Y1101237, LY12F02012, and Y107759).

References

- [1] R. A. Hummel, “Histogram modification techniques,” *Computer Graphics*, vol. 4, no. 3, pp. 209–224, 1975.
- [2] I. M. Bockstein, “Color equalization method and its application to color image processing,” *Journal of the Optical Society of America A*, vol. 3, no. 5, pp. 735–737, 1986.
- [3] D. C. C. Wang, A. H. Vagnucci, and C. C. Li, “Digital image enhancement: a survey,” *Computer Vision, Graphics and Image Processing*, vol. 24, no. 3, pp. 363–381, 1983.
- [4] J. D. Tubbs, “A note on parametric image enhancement,” *Pattern Recognition*, vol. 20, no. 6, pp. 617–621, 1987.
- [5] J. O. Limb, “Distortion criteria of the human viewer,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 12, pp. 778–793, 1979.

- [6] A. B. Watson, *Digital Images and Human Vision*, The MIT Press, Massachusetts, Mass, USA, 1993.
- [7] G. Wyszecki and W. S. Stiles, *Color Science*, Wiley, New York, NY, USA, 1982.
- [8] M. D. Buchanan, "Effective utilization of color in multidimensional data presentations," in *Proceedings of the Seminar Advances in Display Technology*, pp. 9–18, 1979.
- [9] P. E. Trahanias and A. N. Venetsanopoulos, "Color image enhancement through 3-D histogram equalization," in *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, pp. 545–548, 1992.
- [10] W. Niblack, *An Introduction to Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1986.
- [11] R. N. Strickland, C. S. Kim, and W. F. McDonnell, "Digital color image enhancement based on the saturation component," *Optical Engineering*, vol. 26, no. 7, pp. 609–616, 1987.
- [12] A. Toet, "Multiscale color image enhancement," *Pattern Recognition Letters*, vol. 13, no. 3, pp. 167–174, 1992.
- [13] J. Sun, B. Feng, and W. Xu, "Particle swarm optimization with particles having quantum behavior," in *Proceedings of the 2004 Congress on Evolutionary Computation (CEC '04)*, pp. 325–331, June 2004.
- [14] J. Sun, W. Xu, and B. Feng, "A global search strategy of quantum-behaved particle swarm optimization," in *Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems*, pp. 111–116, December 2004.
- [15] J. Sun, W. Fang, X. Wu, V. Palade, and W. Xu, "Quantum-behaved particle swarm optimization: analysis of the individual particle's behavior and parameter selection," *Evolutionary Computation*, vol. 20, no. 3, pp. 349–393, 2012.
- [16] J. Sun, X. Wu, V. Palade, W. Fang, C.-H. Lai, and W. Xu, "Convergence analysis and improvements of quantum-behaved particle swarm optimization," *Information Sciences*, vol. 193, pp. 81–103, 2012.
- [17] J. Sun, X. Wu, W. Fang, Y. Ding, H. Long, and W. Xu, "Multiple sequence alignment using the hidden Markov model trained by an improved quantum-behaved particle swarm optimization," *Information Sciences*, vol. 182, pp. 93–114, 2012.
- [18] J. Sun, W. Fang, V. Palade, X. Wu, and W. Xu, "Quantum-behaved particle swarm optimization with Gaussian distributed local attractor point," *Applied Mathematics and Computation*, vol. 218, no. 7, pp. 3763–3775, 2011.
- [19] J. Sun, W. Chen, W. Fang, X. Wu, and W. Xu, "Gene expression data analysis with the clustering method based on an improved quantum-behaved Particle Swarm Optimization," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 2, pp. 76–391, 2012.
- [20] M. K. Kundu and S. K. Pal, "Automatic selection of object enhancement operator with quantitative justification based on fuzzy set theoretic measures," *Pattern Recognition Letters*, vol. 11, no. 12, pp. 811–829, 1990.
- [21] A. Rosenfeld and A. C. Kark, *Digital Picture Processing*, Academic Press, New York, NY, USA, 1982.
- [22] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the 1995 IEEE International Conference on Neural Networks*, vol. 1944, pp. 1942–1948, December 1995.
- [23] R. C. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the IEEE Conference on Evolutionary Computation*, pp. 81–86, Seoul, Republic of Korea, May 2001.
- [24] P. J. Angeline, "Evolutionary optimization versus particle swarm optimization: philosophy and performance differences," in *Proceedings of the 7th International Conference on Evolutionary Programming (EP '98)*, vol. 1447 of *Lecture Notes in Computer Science*, pp. 601–610, 1998.
- [25] P. J. Angeline, "Using selection to improve particle swarm optimization," in *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 84–89, May 1998.
- [26] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization," in *Proceedings of the 1999 Congress on Evolutionary Computation*, pp. 1951–1957, 1999.
- [27] J. Kennedy, "Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance," in *Proceedings of the 1999 Congress on Evolutionary Computation*, pp. 1931–1938, Washington, DC, USA, 1999.
- [28] J. Kennedy, "Bare bones particle swarms," in *Proceedings of the 2003 IEEE Swarm Intelligence Symposium*, pp. 80–87, 2003.
- [29] J. Kennedy, "Probability and dynamics in the particle swarm," in *Proceedings of the 2004 Congress on Evolutionary Computation (CEC '04)*, pp. 340–347, June 2004.
- [30] J. Kennedy, "In search of the essential particle swarm," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 1694–1701, July 2006.
- [31] F. van den Bergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, 2004.
- [32] S. Janson and M. Middendorf, "A hierarchical particle swarm optimizer and its adaptive variant," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 35, no. 6, pp. 1272–1282, 2005.
- [33] D. Bratton and J. Kennedy, "Defining a standard for particle swarm optimization," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '07)*, pp. 120–127, April 2007.
- [34] F. van den Bergh, *An Analysis of Particle Swarm Optimizers*, University of Pretoria, Pretoria, South Africa, 2001.
- [35] Y. Cai, J. Sun, J. Wang et al., "Optimizing the codon usage of synthetic gene with QPSO algorithm," *Journal of Theoretical Biology*, vol. 254, no. 1, pp. 123–127, 2008.
- [36] W. Chen, J. Sun, Y. Ding, W. Fang, and W. Xu, "Clustering of gene expression data with quantum-behaved particle swarm optimization," in *Proceedings of the 21th International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems*, pp. 388–396, 2008.
- [37] L. D. Coelho, "A quantum particle swarm optimizer with chaotic mutation operator," *Chaos, Solitons and Fractals*, vol. 37, no. 5, pp. 1409–1418, 2008.
- [38] L. dos Santos Coelho, N. Nedjah, and L. de Macedo Mourelle, "Gaussian quantum-behaved particle swarm optimization applied to fuzzy PID controller design," *Studies in Computational Intelligence*, vol. 121, pp. 1–15, 2008.
- [39] L. S. Dos Coelho and P. Alotto, "Global optimization of electromagnetic devices using an exponential quantum-behaved particle swarm optimizer," *IEEE Transactions on Magnetics*, vol. 44, no. 6, pp. 1074–1077, 2008.
- [40] F. Gao, Z.-Q. Li, and H.-Q. Tong, "Parameters estimation online for Lorenz system by a novel quantum-behaved particle swarm optimization," *Chinese Physics B*, vol. 17, no. 4, pp. 1196–1201, 2008.
- [41] S. Li, R. Wang, W. Hu, and J. Sun, "A new QPSO based BP neural network for face detection," *Advances in Soft Computing*, vol. 40, pp. 355–363, 2007.

- [42] S. M. Mikki and A. A. Kishk, "Quantum particle swarm optimization for electromagnetics," *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 10, pp. 2764–2775, 2006.
- [43] S. N. Omkar, R. Khandelwal, T. V. S. Ananth, G. Narayana Naik, and S. Gopalakrishnan, "Quantum behaved particle swarm optimization (QPSO) for multi-objective design optimization of composite structures," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11312–11322, 2009.
- [44] S. L. Sabat, L. dos Santos Coelho, and A. Abraham, "MESFET DC model parameter extraction using quantum particle swarm optimization," *Microelectronics Reliability*, vol. 49, no. 6, pp. 660–666, 2009.
- [45] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

